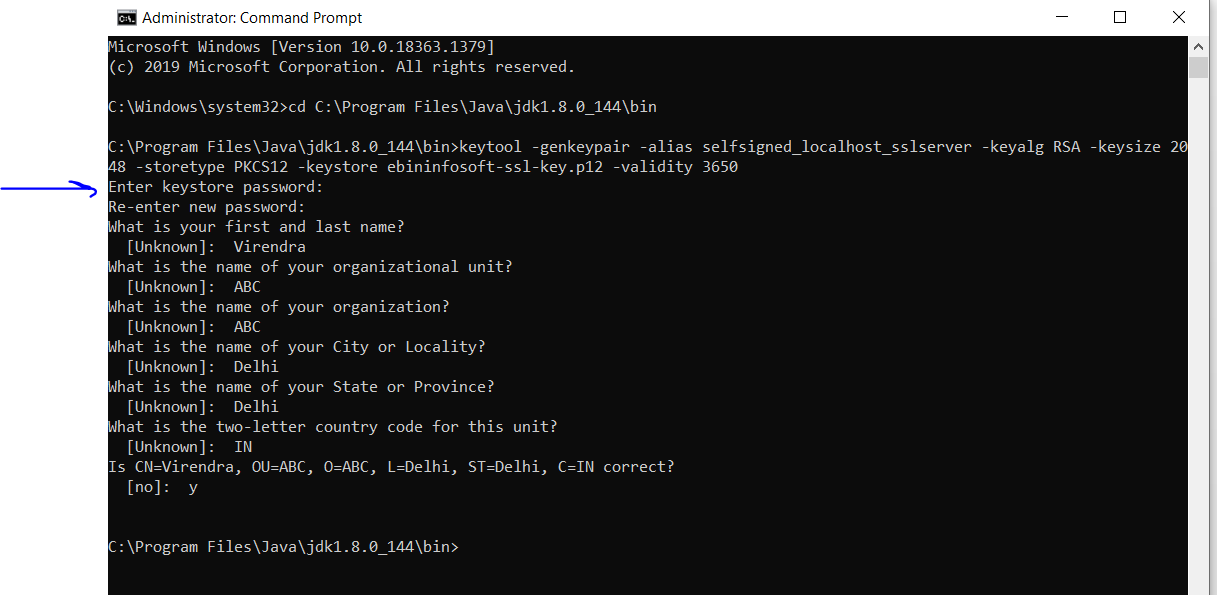
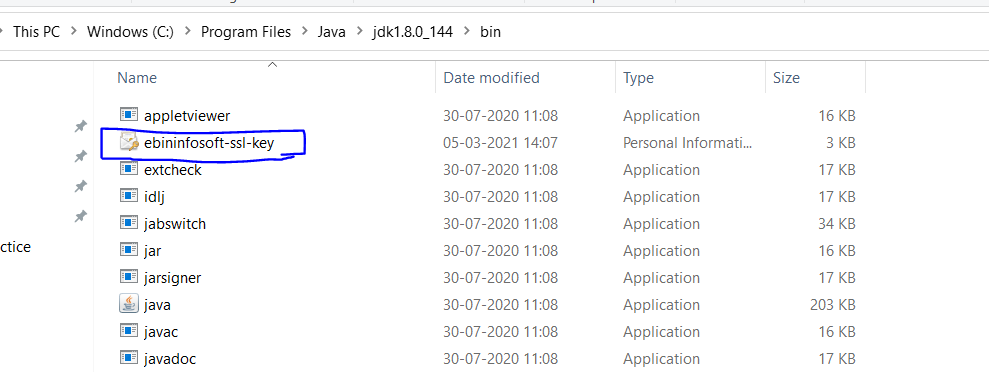
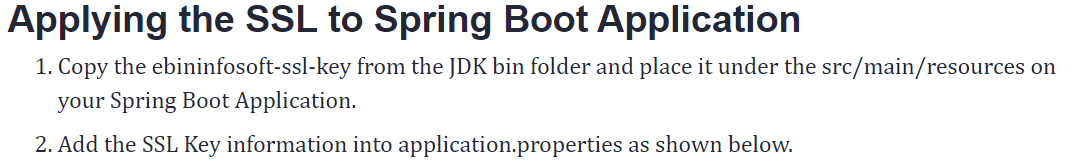
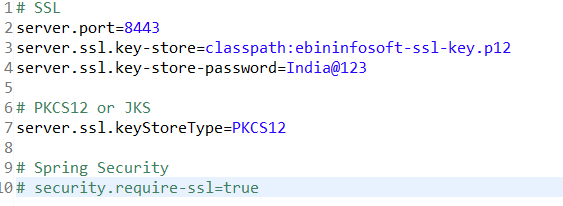
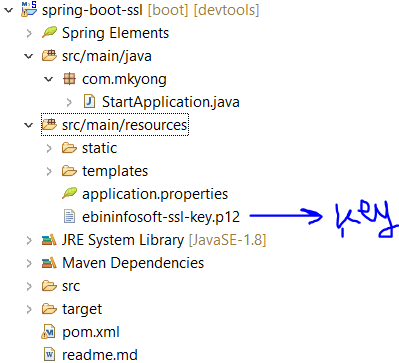
  
1. Open cmd in “Administrator mode”  
2. Go to JDK bin folder path:   
C:\Windows\system32> cd C:\Program Files\Java\jdk1.8.0\_144\bin  
3. Type below keytool command:  
keytool -genkeypair -alias selfsigned\_localhost\_sslserver -keyalg RSA -keysize 2048 -storetype PKCS12 -keystore ebininfosoft-ssl-key.p12 -validity 3650  
4. Enter password:  
5. Re-enter password: India@123  
6. Enter first and last name: Virendra  
7. Enter organizational unit: ABC  
8. Enter name of your organization: ABC  
9. Enter city or locality: Delhi  
10. Enter state or province: Delhi  
11. Enter country code: IN   
  


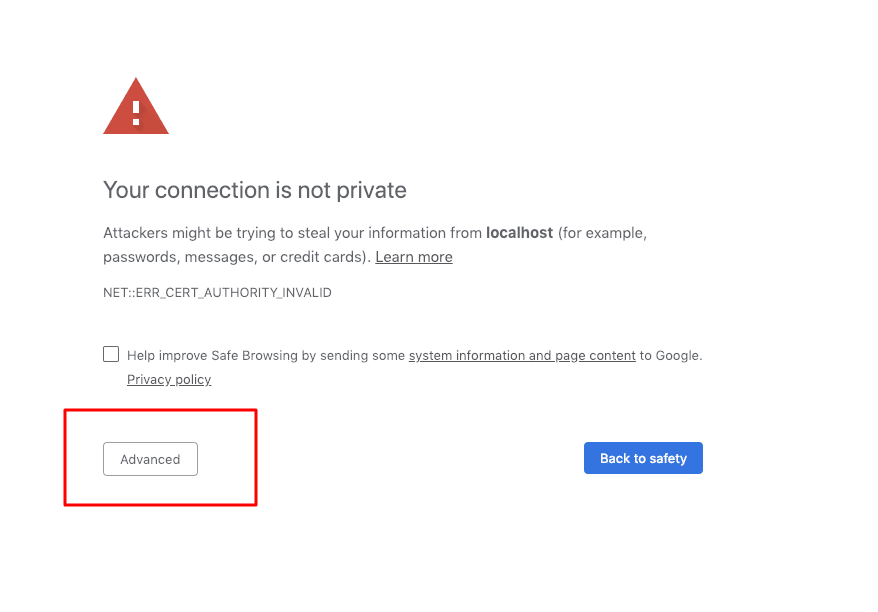
The key will be generated in bin fold India@123er of JDK as shown below:

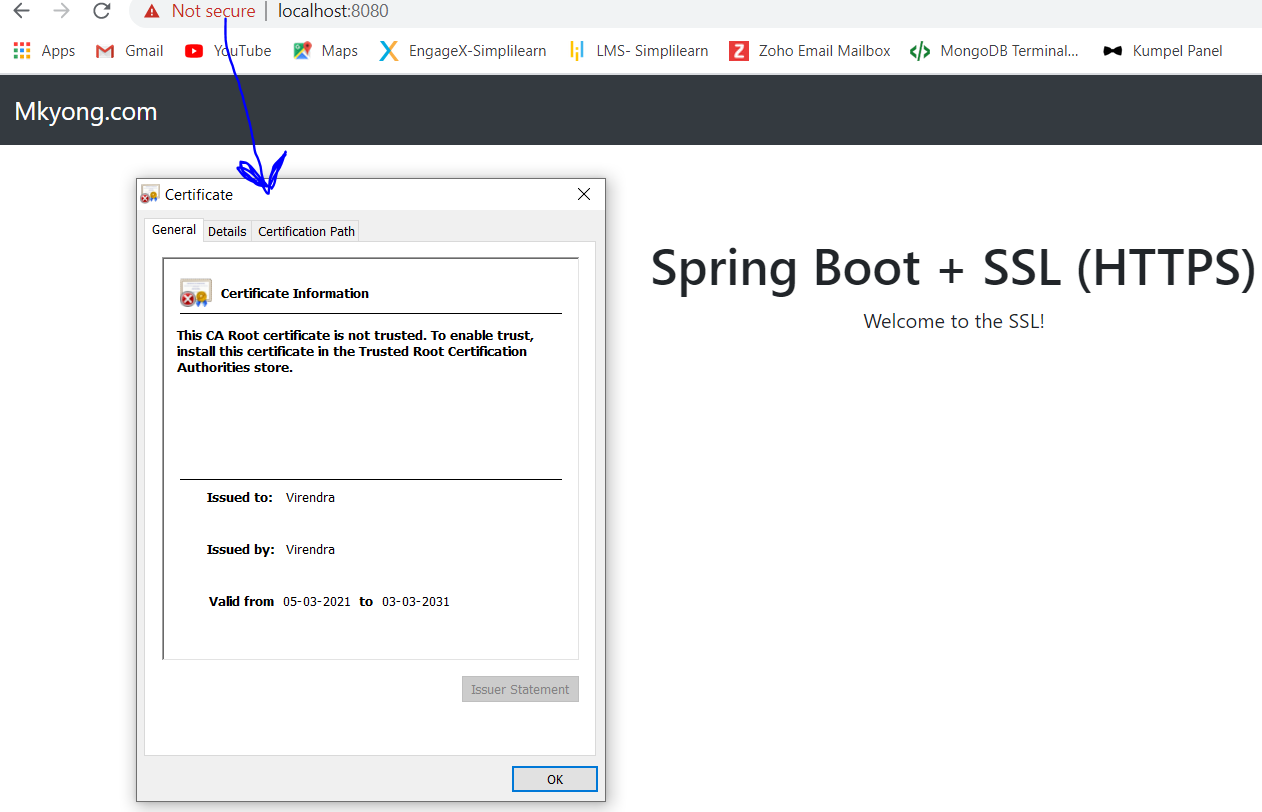
  
  
  
  
With above configuration, we have *enabled HTTPS in Spring Boot application*. Our application is ready to accept and server over HTTPS.

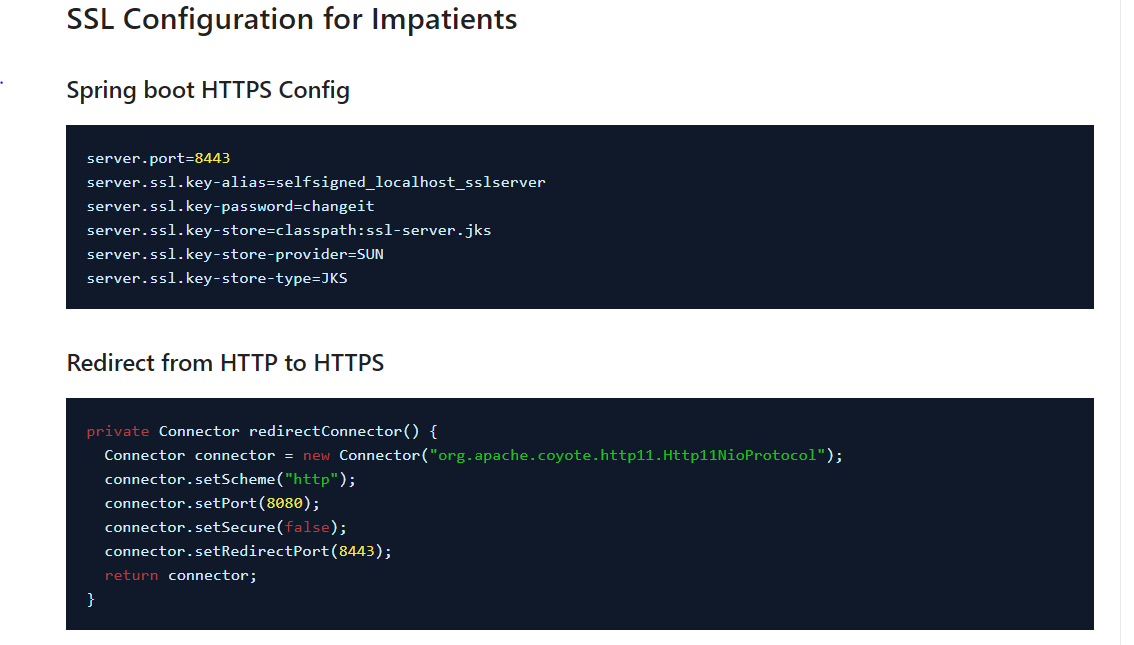


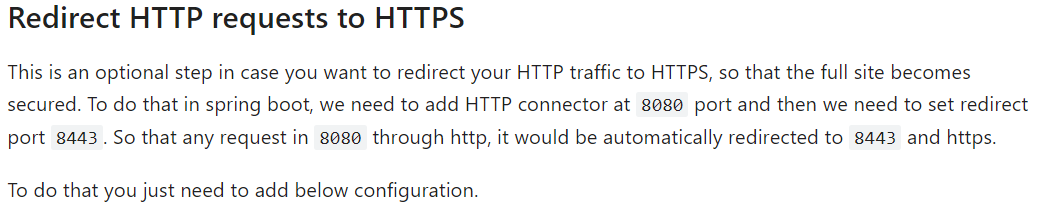
Copy the key generated under the folder 🡪 /src/main/resources

Run the application 🡪 StartApplication.java  
Open the browser 🡪 <https://localhost:8443/>







  
  
@Bean

public EmbeddedServletContainerFactory servletContainer() {

  TomcatEmbeddedServletContainerFactory tomcat = new TomcatEmbeddedServletContainerFactory() {

      @Override

      protected void postProcessContext(Context context) {

        SecurityConstraint securityConstraint = new SecurityConstraint();

        securityConstraint.setUserConstraint("CONFIDENTIAL");

        SecurityCollection collection = new SecurityCollection();

        collection.addPattern("/\*");

        securityConstraint.addCollection(collection);

        context.addConstraint(securityConstraint);

      }

    };

  tomcat.addAdditionalTomcatConnectors(redirectConnector());

  return tomcat;

}

private Connector redirectConnector() {

  Connector connector = new Connector("org.apache.coyote.http11.Http11NioProtocol");

  connector.setScheme("http");

  connector.setPort(8080);

  connector.setSecure(false);

  connector.setRedirectPort(8443);

  return connector;

}

